CLAIMS

What is claimed is:

1. A method of assigning Walsh codes comprising the steps of:

(a) receiving as input a status vector for a Walsh code system of length 2^n;

(b) creating a new status vector for a selected Walsh code length of $j = 2^{n-k}$ from the status vector;

(c) creating a search mask for the selected Walsh code length of j;

(d) creating a search sequence for the selected Walsh code length of j; and

(e) searching the search sequence with the search mask to find the next available Walsh code.

2. The method of claim 1 wherein step (b) comprises the steps of:

(b1) copying the status vector to a new status vector for the desired Walsh code length j;

(b2) initializing a loop index k to zero;

(b3) incrementing the loop index k by one;

(b4) replacing the new status vector with the new status vector OR'd with the new status vector shifted right by 2^{n-k} bits; and

(b5) repeating steps (b3) and (b4) until 2^{n-k} equals the desired Walsh code length j.

3. The method of claim 1 wherein step (e) comprises the steps of:

(e1) shifting the search mask left by a number of bits corresponding to a next search sequence entry M to generate a shifted search mask;

(e2) performing an AND operation between the shifted search mask and the new status vector; and

(e3) generating as output a Walsh code M of length j if the result of step (e2) equals zero.

4. The method of claim 3 further comprising the steps of:

(e5) returning to step (e1) if the search sequence entry M is not last in the search sequence and if the result of step (e2) equals the search mask; and

(e6) generating as output a null Walsh code indicating that no Walsh code is available at the selected length j if M is last in the search sequence.

5. The method of claim 4 further comprising the steps of:

(e7) creating a new search mask for a Walsh code of the selected length j if the result of step (e2) does not equal the search mask;

(e8) shifting the new search mask left by a number of bits corresponding to the search sequence entry M to generate a shifted search vector;

(e9) performing an AND operation between the shifted search vector and the new status vector; and

(e10) generating as output a Walsh code M of length j if the result of step (e9) equals zero.

6. The method of claim 5 further comprising the step of (e11) generating as output a Walsh code $M + 2^{n-k}$ of length j if the result of step (e9) does not equal zero.

7. A method of tracking an assignment status of each Walsh code in a Walsh code system comprising the steps of:

(a) receiving as input a status vector, an assignment indicator, a Walsh code parameter *M*, and a Walsh code length parameter *j* wherein *M* and *j* are positive integers;

(b) retrieving a bit mask [M,j]; and

(c) updating the status vector as a function of the Walsh code parameter *M*, the assignment indicator, and the bit mask [M,j].

8. The method of Claim 7 wherein step (c) comprises the following steps:

(c1) checking whether the assignment indicator indicates an assignment or a release of Walsh code *M* of length *j*;

(c2) performing an OR operation between the status vector and the bit mask [M,j] if the assignment indicator indicates an assignment; and

(c3) replacing the status vector with a result of the OR operation between the status vector and the bit mask [M,j] to set covered Walsh codes in the status vector.

9. The method of Claim 7 wherein step (c) comprises the following steps:

(c1) performing a negation operation on the bit mask [M,j] if the assignment indicator indicates a release;

(c2) performing an AND operation between the status vector and the result of the negation operation; and

(c3) replacing the status vector with a result of the AND operation between the status vector and the result of the negation operation to clear uncovered Walsh codes in the status vector.

10. A computer program product comprising:

a medium for embodying a computer program for input to a computer; and

a computer program embodied in the medium for causing the computer to perform the following functions:

(a) receiving as input a status vector for a Walsh code system of length 2^n;

(b) creating a new status vector for a selected Walsh code length of $j = 2^{n-k}$ from the status vector;

(c) creating a search mask for the selected Walsh code length of j;

(d) creating a search sequence for the selected Walsh code length of j; and

(e) searching the search sequence with the search mask to find an available Walsh code.

11. The computer program product of claim 10 wherein step (b) comprises the steps of:

(b1) copying the status vector to a new status vector for the desired Walsh code length j;

(b2) initializing a loop index k to zero;

(b3) incrementing the loop index k by one;

(b4) replacing the new status vector with the new status vector OR'd with the new status vector shifted right by 2^{n-k} bits; and

(b5) repeating steps (b3) and (b4) until 2^{n-k} equals the desired Walsh code length j.

12. The computer program product of claim 10 wherein step (e) comprises the steps of:

(e1) shifting the search mask left by a number of bits corresponding to a next search sequence entry M to generate a shifted search mask;

(e2) performing an AND operation between the shifted search mask and the new status vector; and

(e3) generating as output a Walsh code *M* of length *j* if the result of step (e2) equals zero.

13. The computer program product of claim 12 further comprising the steps of:

(e5) returning to step (e1) if the search sequence entry *M* is not last in the search sequence and if the result of step (e2) equals the search mask; and

(e6) generating as output a null Walsh code indicating that no Walsh code is available at the selected length *j* if the search sequence entry *M* is last in the search sequence.

14. The computer program product of claim 13 further comprising the steps of:

(e7) creating a new search mask for a Walsh code of the selected length *j* if the result of step (e2) does not equal the search mask;

(e8) shifting the new search mask left by a number of bits corresponding to the search sequence entry *M* to generate a shifted search vector;

(e9) performing an AND operation between the shifted search vector and the new status vector; and

(e10) generating as output a Walsh code *M* of length *j* if the result of step (e9) equals zero.

15. The computer program product of claim 14 further comprising the step of (e11) generating as output a Walsh code $M + 2^{n-k}$ of length *j* if the result of step (e9) does not equal zero.

16. A computer program product comprising:

a medium for embodying a computer program for input to a computer; and

a computer program embodied in the medium for causing the computer to perform the following functions:

(a) receiving as input a status vector, an assignment indicator, a Walsh code parameter *M*, and a Walsh code length parameter *j* wherein *M* and *j* are positive integers;

(b) retrieving a bit mask [M,j]; and

(c) updating the status vector as a function of the Walsh code parameter *M*, the assignment indicator, and the bit mask [M,j].

17. The computer program product of Claim 16 wherein step (c) comprises the following steps:

(c1) checking whether the assignment indicator indicates an assignment or a release of Walsh code *M* of length *j*;

(c2) performing an OR operation between the status vector and the bit mask [M,j] if the assignment indicator indicates an assignment; and

(c3) replacing the status vector with a result of the OR operation between the status vector and the bit mask [M,j] to set covered Walsh codes in the status vector.

18. The computer program product of Claim 16 wherein step (c) comprises the following steps:

(c1) performing a negation operation on the bit mask [M,j] if the assignment indicator indicates a release;

(c2) performing an AND operation between the status vector and the result of the negation operation; and

(c3) replacing the status vector with a result of the AND operation between the status vector and the result of the negation operation to clear uncovered Walsh codes in the status vector.